2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2025)
October 19-25, 2025. Hangzhou, China

2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 979-8-3315-4393-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/IROS60139.2025.11247615

Recognizing Skeleton-Based Actions As Points

Baiqiao Yin[1,2], Jiaying Lin[1,2], Jiajun Wen[3], Yue Li[3], Jinfu Liu[3], Yanfei Wang[3] and Mengyuan Liu[1,2,†]

***Abstract*— Recent advances in skeleton-based action recognition have been primarily driven by Graph Convolutional Networks (GCNs) and skeleton transformers. While conventional approaches focus on modeling joint co-occurrences through skeletal connections, they overlook the inherent positional information in 3D coordinates. Although Hyper-graphs partially address the limitation of pairwise aggregation in capturing higher-order kinematic dependencies, challenges remain in their topological definitions. To solve these problems, this paper proposes a skeleton-to-point network (Skeleton2Point) to model joints' position relationships directly in three-dimensional space without fixed topology limitation, which is the first to regard skeleton recognition as point clouds. However, simply considering the raw 3D coordinates would result in the loss of the anatomical identity of each keypoint and its temporal position in the sequence. To address this limitation, we augment the three-dimensional spatial coordinates with two additional dimensions: the anatomical index of each keypoint and its corresponding frame number with a proposed Information Transform Module (ITM). This transformation extends the representation from a three-dimensional to a five-dimensional feature space. Furthermore, we propose a Cluster-Dispatch-based Interaction module (CDI) to enhance the discrimination of local-global information. In comparison with existing methods on NTU-RGB+D 60 and NTU-RGB+D 120 datasets, Skeleton2Point has demonstrated state-of-the-art performance on both joint modality and stream fusion. Especially, on the challenging NTU-RGB+D 120 dataset under the X-Sub and X-Set setting, the accuracies reach 90.63% and 91.92%.**

I. Introduction

Human action recognition is an extremely important task in the field of computer vision, which also has great research value and broad application prospects in education [1], human-computer interaction [2], and content-based video retrieval [3]. For human action videos, various modalities derived from rich multimedia are beneficial to the recognition task, including RGB, optical flow, and human skeletons. Among them, skeleton-based action recognition algorithms have attracted many researchers due to their robustness against the variation of appearance and background. Skeleton data contain 3D human joint coordinates and their connection matrix, representing information about the joint's position and the bone connections between them.

†Corresponding author: Mengyuan Liu (liumengyuan@pku.edu.cn)
[1]State Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School.
[2]Guangdong Provincial Key Laboratory of Ultra High Definition Immersive Media Technology, Peking University, Shenzhen Graduate School.
[3]Sun Yat-sen University.

This work was supported by National Natural Science Foundation of China (No. 62473007), Guangdong Provincial Key Laboratory of Ultra High Definition Immersive Media Technology (No. 2024B1212010006), Shenzhen Innovation in Science and Technology Foundation for The Excellent Youth Scholars (No. RCYX20231211090248064).

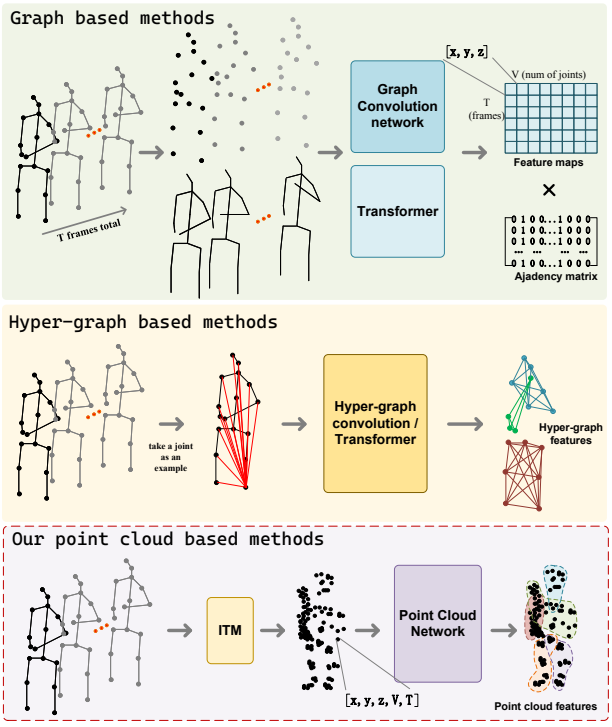


Fig. 1: Conventional GCN-based and Hyper-graph-based methods heavily rely on dense graphs to build the relations among joints. However, the 3d coordinate channels also contain the relative positional information in 3d space, which almost all existing works neglected. By treating the joint coordinates as a point cloud, we could model the joints' position relations in 3d space.

A typical way to use skeletons for action recognition is to build Graph Convolutional Networks (GCNs). Since [4] proposed STGCN to model skeletal data as a spatio temporal graph structure, graph convolutional networks have developed rapidly. The joints and bones in the human body naturally form graphs, which makes GCNs a perfect tool for extracting topological features of skeletons. More recently, a limitation of GCNs has been identified, i.e., the topology is fixed after training. To relax such a restriction, the Self-Attention (SA) mechanism has been adopted to make the topology of GCNs adaptive to the input, resulting in the state-of-the-art hybrid models ([5][6]). There are also several works like [7][8] that leverage hyper-graph to reveal the underlying issue of graph models in general, i.e., pairwise aggregation essentially ignores the high-order kinematic dependencies between body joints. However, these all methods are limited to the modeling of topology structure between joints, heavily rely on dense graphs to build the relations among joints, **neglecting to model the implicit relative**

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 06:21:34 UTC from IEEE Xplore. Restrictions apply.

positional information contained in the 3D coordinates channel itself, as illustrated in Fig 1.

To solve the above problem, we propose a skeleton-to-point network (Skeleton2Point) that consists of two trunk branches. The human skeletal structure is represented by several keypoints per frame, each with three-dimensional coordinates, across multiple temporal frames. Rather than treating these keypoints conventionally, we propose viewing the entire sequence as a unified point cloud. However, simply considering the raw 3D coordinates would result in the loss of crucial information - specifically, the anatomical identity of each keypoint and its temporal position in the sequence. To address this limitation, we augment the three-dimensional spatial coordinates with two additional dimensions: the anatomical index of each keypoint and its corresponding frame number. Next, FPS and kNN are used to sample and model the position relations between the points, then a point cloud information extractor is leveraged to extract latent features. We also propose a Cluster-Dispatch-based interaction module to enhance the discrimination of local-global information. The results from both branches are integrated to make final predictions. By leveraging the proposed Skeleton2Point, we better capture spatial relationships between joints to achieve better human action recognition.

Our main contributions can be summarized as follows:

- To our best knowledge, we are the first to regard skeleton joints as point clouds while incorporating the spatial and temporal information into point cloud methods, demonstrating the validity of modeling position relationships with 3D coordinates.
- We devise a novel information transformation module (ITM) to merge the original time and series information and the joint coordinates information. We also propose a Cluster-Dispatch-based interaction module (CDI) to focus on overall movement trends.
- We conduct extensive experiments on NTU-RGB+D 60 and NTU-RGB+D 120 datasets to compare our proposed method with the state-of-the-art models in the joint stream and multi-streams. Experimental results demonstrate the significant improvement of our method. In the most challenging NTU120_XSub&XSet, our method achieves sota by a large margin.

II. Related works

A. Skeleton Based Human Action Recognition

Inspired by the development of Graph Convolutional Networks (GCN), Spatial-Temporal GCNs are used to extract high-level features from skeletons since the joints and bones in the human body naturally construct a graph. An early application of GCN on skeleton-based action recognition is STGCN [4], which uses stacked GCN blocks to process skeleton data, while each block consists of a spatial module and a temporal module. [9] proposed a channel-wise topology graph convolution, which models channel-wise topology with a refinement method.

Recently, transformers have also been explored in skeleton-based action recognition. [5] and [10] unified spatial and temporal modeling within the transformer via segment temporal aggregation and physical connectivity constraints in which way the topology of the human body is fully exploited. [11] is proposed with comprehensive high-performance spatial temporal attention design and topological information fusion. Nevertheless, due to the lack of a module focusing on position information extracting, the methods above couldn't achieve excellent local-global positional relationship capture.

B. Point Cloud Classification

The major method for processing the point clouds is point-based modeling. [12] is a pioneering work that successfully applies deep architecture on raw point sets, with shared multi-layer perceptrons (MLP) used. PointNet++ [13] is built on top of PointNet [12], which learns hierarchical point cloud features and can aggregate features in local geometric neighbors. Recently, PointNeXt [14] explored more advanced training and data augmentation strategies with the PointNet++ backbone to further improve accuracy and efficiency. Point Transformer proposes a modified Transformer architecture that aggregates local features with vector attention and relative position encoding [15]. Following them, some works have extended the point-based methods to various local aggregation operators such as PointConT [16], achieving a sota of categorization tasks by combining various outstanding modules in point-based modeling. Point cloud models show excellent performance in learning 3D coordinate information, and the joint modality of skeleton data also consists of 3D coordinate information of each joint. So naturally, we think of using point cloud models to deep-dive into the position relationships of skeleton joints. However, there is no similar exploration in existing works, so we propose Skeleton2Point, which learns the skeleton joints' feature along with three-dimensional position information using point cloud methods.

III. Method

A. Human Skeleton Learning

Backbone. Transformer-based methods have achieved success in skeleton-based action recognition due to their unique advantages in modeling joint relations. Our Skeleton2Point also embraces graph transformer as the backbone to model skeleton features. The input of our model is a sequence of skeletons with a shape of $T \times V \times 3$, which means T frames of V joints in a 3D space. We build our approach on [11]. The backbone consists of 10 basic graph transformers. A graph transformer is typically composed of a spatial transformer and a temporal transformer. The normal spatial transformer utilizes the attention matrix A_t and adjacency matrix A_i for aggregate features of neighbor vertices to update the features f_i, which can be expressed as follows:

$$f_i^{\text{out}} = f_i^{\text{in}} + V_i A, \tag{1}$$

$$V_i = \text{Conv}_{1\times1}\left(\text{split}_n\left(\text{trans}_v\left(f_i^{\text{in}}\right)\right)\right). \tag{2}$$

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 06:21:34 UTC from IEEE Xplore. Restrictions apply.

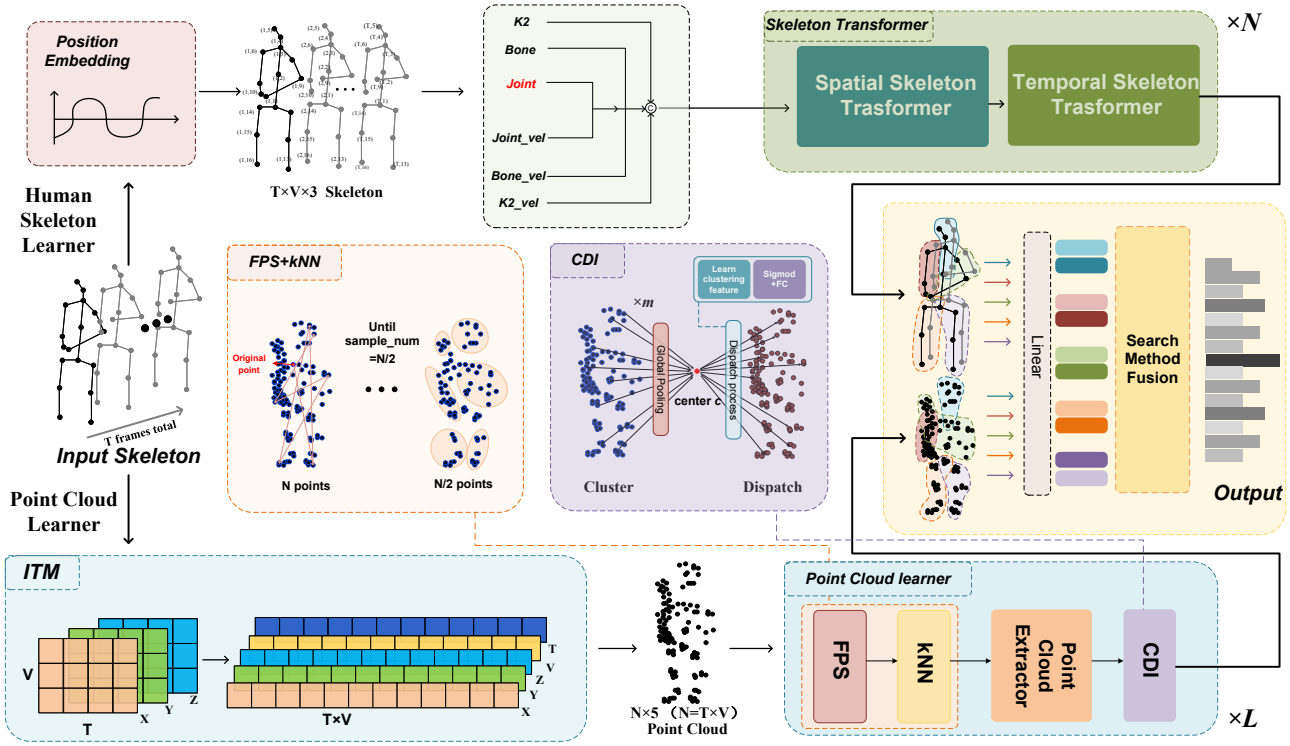


Fig. 2: Framework of our proposed Skeleton2Point. It leverages a dual-learner approach. The human skeleton learner feeds compact skeletal representations in the skeleton transformer network, which is composed of a spatial transformer block and a temporal transformer block. In the point cloud learner, skeleton data is transformed into point cloud's form with a proposed Information Transform Module (ITM), which fills the channel information with the spatial and temporal serial number. Then, several point cloud learning layers are adopted to extract deep position features. The point cloud learning level is made of three key layers: Sampling layer, Grouping layer, and Point cloud extract layer. We also use a Cluster-Dispatch-based Interaction module (CDI) to enhance the discrimination of local-global information. Finally, we adopt a weight search algorithm based on Gaussian Process Bayesian Optimization for the final fusion.

where A can be defined as the attention matrix A_t, static (defined manually) adjacency matrix A_I and dynamic adjacency matrix A_d (initialized manually but learnable). The Eq 2 denotes the multi-head mechanism in self-attention that our model used and the attention matrix A_t is calculated as follows:

$$Q, K = \sigma\left(\text{linear}\left(\text{pool}\left(\text{split}\left(f\right)\right)\right)\right), \tag{3}$$

$$A_t = \text{softmax}\left(\text{atten}\left(Q, K\right)\right). \tag{4}$$

Our Skeleton2Point feeds the joint modality skeleton into ten dynamic graph transformer blocks for feature extraction. Multi-stream strategy [11] is specifically used for residual information to extract key information twice for the final classification.

B. Point Cloud Learning

From Skeleton to Point Clouds. In the point cloud branch, we propose the information transform module (ITM) to transform the skeleton data into point cloud's form as shown in Fig.3. Given an input skeleton $\mathbf{S} \in R^{3\times T\times V}$, we begin by filling the channel information with the spatial and temporal serial numbers of each joint $\mathbf{S}_{i,j}$, where each joint's serial number is presented as $\left[\frac{i}{T} - 0.5, \frac{j}{V} - 0.5\right]$. The improved skeleton is then converted to a collection of point clouds $\mathbf{P} \in R^{5\times N}$, where $n = T \times V$ is the number of points, and each point contains both position (coordinates) and spatial-temporal serial information.

Point Cloud Set Feature Learning. After modality transformation, several point cloud learning levels are adopted to extract deep position features. The point cloud learning level is made of three key layers: Sampling layer, Grouping layer, and Point cloud extract layer. The Sampling layer selects a set of points from input points, which defines the centroids of local regions. The grouping layer then constructs local region sets by finding "neighboring" points around the centroids. The point cloud extract layer uses [16] as lite-backbone and [17] as heavy-backbone to encode local region patterns into feature vectors. We introduce the layers of a point cloud learning level in the following paragraphs.

- **Sampling layer.** Given input points $\{x_1, x_2, \ldots, x_n\}$,we use iterative farthest point sampling (FPS) to choose a subset of points $\{x_{i_1}, x_{i_2}, \ldots, x_{i_m}\}$, such that x_{i_j} is the most distant point with $\{x_{i_1}, x_{i_2}, \ldots, x_{i_j-1}\}$regard to the rest points. Compared with random sampling, it has better coverage of the entire point set given the same number of centroids.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 06:21:34 UTC from IEEE Xplore. Restrictions apply.

- **Grouping layer.** We use k-nearest neighbor (kNN) for grouping. The input to this layer is a point set of size $N \times (d + C)$ and the coordinates of a set of centroids of size $N' \times d$. The output is groups of point sets of size $N' \times K \times (d + C)$, where each group corresponds to a local region and K is the number of points in the neighborhood of centroid points. Note that K varies across groups but the succeeding Point cloud extract layer can convert a flexible number of points into a fixed-length local region feature vector and we set the number K to 24.
- **Point cloud extract layer.** In this layer, the input are N' local regions of points with data size $N' \times K \times (d+C)$. Each local region in the output is abstracted by its centroid and local feature that encodes the centroid's neighborhood. Output data size is $N' \times (d + C')$. We choose PointConT [16] as the lite building block and PointMLP [17] as the heavy building block for local pattern extracting. It is noted that the detailed implementation of the block is not the main concern of our method. The implementation of the point cloud extract unit can be replaced by any other point cloud network module.

Cluster-Dispatch based interaction module. Inspired by the [18], we also propose a Cluster-Dispatch-based interaction module (CDI) to make the point cloud backbone focus on the overall movement trends. Suppose an action sample contains m points and the center is c, all m points in the sample are aggregated by global averaging to get the center point. Assuming the similarity between the m points and the center is $s \in R^m$, we obtain $p_v \in R^{m \times d'}$ by mapping these m points to the value space, where d' is the value dimension. Similarly, there is a clustering center c_v in the value space and the clustering feature $f \in R^{d'}$ can be written as:

$$\begin{aligned} f &= \frac{1}{\gamma}\left(\sigma\left(\sum_{i=1}^{m} \alpha s_i + \beta\right) \cdot p_v + c_v\right), \\ &\text{s.t., } \gamma = 1 + \sum_{i=1}^{m} \sigma\left(\alpha s_i + \beta\right). \end{aligned} \tag{5}$$

Here α and β are learnable scalars to scale and shift the similarity. $\sigma()$ is a sigmoid function to re-scale the similarity to (0, 1) and s_i denotes the similarity between the i-th point and the center. To dispatch the feature, the aggregated features f are adaptively assigned to each point in the action sample, allowing the points to communicate with each other and share features from all points by:

$$p_i' = p_i + \text{FC}(\text{sig}(\alpha s_i + \beta) \cdot f). \tag{6}$$

In the original space, p_i denotes the i-th point and p_i' denotes the point after reallocation. Ablation experiments in Table I. demonstrate the effectiveness of the proposed modules.

C. Gaussian Search Method based Fusion

We validate the model under the joint modality only and 6-stream fusion (6s), respectively. The joint modality only is used because joints contain 3D coordinates same as point clouds, so we want to verify that the deep position feature that point cloud learning had extracted would help the skeleton backbone learn the joint position complementarily. Following prior work [19], the input of multiple streams refers to $\tilde{\mathcal{X}}_k = \left(I - P^k\right)\mathcal{X}$, where $k = 1, 2, \ldots, K$, and K is set to 2 in NTU-RGB+D 60 dataset.

In a fusion of skeleton and point clouds, fixed parameters limit the performance of each model and make it difficult to achieve optimal results. We follow the weight search algorithm [20] based on Gaussian Process Bayesian Optimization. This algorithm effectively finds the best solution by constructing a probabilistic model of the objective function. Given an objective function $f(x)$ and initial samples $\mathcal{P} = \{(x_i, y_i)\}_{i=1}^{N}$, where x_i represents the input and $y_i = f(x_i)$ represents the observed output. The goal is to find the global optimum x of $f(x)$ within the search space $\mathcal{X}$. The iteration continues until convergence, determined by a predefined number of iterations or convergence criteria.

TABLE I: Ablation studies of point cloud branch on the NTU-RGB+D 60 and NTU-RGB+D 120 datasets with the joint input modality in lite point cloud branch.

Point Extractor	ITM	CDI (parallel)	CDI (cascade)	NTU-RGB+D 60 X-Sub	NTU-RGB+D 120 X-Sub
Lite	✗	✗	✗	81.22	69.35
Lite	✓	✗	✗	**84.11**$^{\uparrow 2.89}$	**79.14**$^{\uparrow 9.79}$
Lite	✓	✓	✗	**84.62**$^{\uparrow 3.40}$	**79.67**$^{\uparrow 10.32}$
Lite	✓	✗	✓	**84.75**$^{\uparrow 3.53}$	**80.17**$^{\uparrow 10.82}$

TABLE II: Ablation studies of point cloud branch on the NTU-RGB+D 60 dataset with the joint input modality in heavy point cloud branch.

Point Extractor	ITM	CDI	NTU-RGB+D 60(X-Sub)
Heavy	✗	✗	81.39
Heavy	✓	✗	**88.17**$^{\uparrow 6.78}$
Heavy	✓	✓	**88.56**$^{\uparrow 7.17}$

IV. Experiments

A. Datasets

NTU RGB+D 60[36] is a large-scale human action recognition dataset collected in an indoor environment, containing 56,880 skeleton action sequences. **NTU RGB+D 120** [37] is currently the largest dataset with 3D joints annotations for human action recognition, which extends NTU RGB+D with an additional 57,367 skeleton sequences over 60 extra action classes.

B. Implementation Details

We follow previous work [9] to process the two datasets and all skeleton sequences are padded to 64 frames. We used Stochastic Gradient Descent (SGD) as the optimizer and cross-entropy as the loss function. On the skeleton branch, we adopt [11] as the human skeleton backbone. In the first 5 epochs, we apply a warmup strategy for stable training. The initial learning rate is set to 0.1 and we decrease it at epoch

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 06:21:34 UTC from IEEE Xplore. Restrictions apply.

TABLE III: Recognition accuracy comparison against state-of-the-art methods on NTU-RGB+D 60 and NTU-RGB+D 120 datasets under the joint and multi-stream modality. Bold text denotes optimal performance, dagger marks † indicate second-best.

Type	Methods	Publisher	NTU-RGB+D 60 Joint		NTU-RGB+D 120 Joint		NTU-RGB+D 60		NTU-RGB+D 120	
			X-Sub(%)	X-View(%)	X-Sub(%)	X-Set(%)	X-Sub(%)	X-View(%)	X-Sub(%)	X-Set(%)
GCN	Shift-GCN[21]	*CVPR'20*	87.80	95.10	80.90	83.20	90.70	96.50	85.90	87.60
	MS-G3D[22]	*CVPR'20*	88.77	94.88	82.35	84.14	91.50	96.20	86.90	88.40
	CTR-GCN[9]	*ICCV'21*	88.95	90.23	84.95	86.68	92.40	96.80	88.90	90.60
	MSTGCN[23]	*AAAI'21*	89.00	95.10	82.80	84.50	92.30	91.50	87.50	88.80
	EfficientGCN[24]	*TPAMI'22*	-	-	-	-	92.10	96.10	88.70	88.90
	Info-GCN[19]	*CVPR'22*	89.80	95.20	85.10	86.30	93.00	97.10	89.80	91.20
	FR-Head[25]	*CVPR'23*	90.33	95.26	85.51	87.32	92.80	96.80	89.50	90.90
	KoompanPool.[26]	*CVPR'23*	90.20	95.20	85.70	87.40	92.90	96.80	90.00	91.30
	HD-GCN[27]	*ICCV'23*	90.60	95.70	85.70	87.30	93.40	97.50†	89.20	91.10
	DS-GCN[28]	*AAAI'24*	-	-	-	-	93.10	97.20	90.10	91.60
	BlockGCN[29]	*CVPR'24*	90.90	95.40	86.90	88.20	93.10	97.00	90.30	91.50
	MMCL[30]	*ACMMM'24*	-	-	-	-	93.50†	97.40	90.30	91.70
Hypergraph	Hyper-GNN[31]	*TIP'21*	-	-	-	-	89.50	95.70	-	-
	DHGNN[32]	*CoRR'21*	-	-	-	-	90.70	96.00	86.00	87.90
	Selective-HCN[8]	*ICMR'21*	-	-	-	-	90.80	96.60	-	-
	SD-HGCN[7]	*ICONIP'21*	-	-	-	-	90.90	96.70	87.00	88.20
Transformer	STF[5]	*AAAI'22*	91.34	96.46	85.06	86.40	92.47	96.86	88.85	89.92
	ST&ST[33]	*ACMMM'23*	90.90	95.40	85.80	87.90	93.10	96.70	89.80	91.20
	TranSkeleton[10]	*TCSVT'23*	-	-	-	-	92.80	97.00	89.40	90.50
	JT-GraphFormer[34]	*AAAI'24*	91.80†	**96.80**	87.60	89.60	93.40	97.50	89.90	91.70
	FreqMixFormer[35]	*ACMMM'24*	91.50	96.00	87.90	89.10	**93.60**	97.40	90.50	91.90†
	Baseline(SkeMixFormer)	-	90.73	95.76	85.82	87.64	93.20	97.20	90.20	91.50
	Skeleton2Point(Lite)	-	91.50	96.28	88.62†	89.92†	93.32	97.35	90.50†	91.76
	Skeleton2Point(Heavy)	-	**92.07**	96.70†	**88.93**	**90.34**	93.41	**97.51**	**90.63**	**91.92**

TABLE IV: Comparison of parameter, computation cost when training&inferring and accuracy on the NTU-RGB+D 120 cross-subject Protocols.

Methods	Publisher	Modality	Param	Flops	NTU120
DRDIS[38]	*TCSVT'21*	Depth+RGB	171.98M	27.92G	81.30
VPN[39]	*ECCV'20*	Ske+RGB	24.00M	-	86.30
MMNet[40]	*TPAMI'22*	Ske+RGB	14.40M	19.2G	88.70
EPPNet[41]	*TRIT24*	Ske+Parsing	25.27M	7.84G	86.70
STAR-Transformer[42]	*WACV24*	Ske+RGB	58.49M	18.67G	90.30
Skeleton2Point(Lite)	*2024*	Ske+PC	16.98M	5.78G	**90.20**
Skeleton2Point(Heavy)	*2024*	Ske+PC	25.87M	10.27G	**90.63**

TABLE V: Comparison of different sampling and grouping methods in NTU-RGB+D 60 under the X-Sub setting.

Sampling	Grouping	Acc(%)
Random sample	kNN	82.43
FPS	Ball(R=0.1)	80.12
FPS	Ball(R=0.2)	81.94
FPS	kNN	**84.75**$^{\uparrow 2.32}$

35 and 55 with a factor of 0.1. We train all models with 90 epochs and select the best performance with a batch size of 128. On the point cloud branch, we adopt [16] as the lite point cloud information extractor and [11] as the heavy point cloud information extractor. The training epoch and learning rate are set to 250 and 0.01 respectively, while the batch size is also set to 128. In the skeleton branch, the base channel C is set to 80, and the hidden channel C_h is set to 320. In the point cloud branch, the base channel C is set to 64, and the hidden channel C_h is set to 1024. Please refer to our code for more details.

C. Ablation Study

Ablation experiments of different components. In Table I and II, we experimentally validate the importance of the components in both lite point cloud branch and heavy point cloud branch, where CDI (parallel) denotes the CDI module is paralleled with the MLP in point cloud information extractor and CDI (cascade) denotes the CDI module is cascaded with the MLP in point cloud information extractor. Table II shows that the branch without ITM has a performance drop of 2.9% and 9.8% on the NTU-RGB+D 60 dataset under the X-Sub setting and NTU-RGB+D 120 dataset under the X-Sub setting respectively. The result drops by 0.64% without using the CDI, which shows that the position modeling needs to interact with global-local information and focus on overall movement trends. In addition, the result of using cascade CDI outcompeting using parallel CDI by a margin of 0.3% reflects the cascade structure's advantage and is more beneficial for supporting position modeling with point cloud methods.

Parameter and computation cost. Additionally, we showcase the parameter and computation cost required for the proposed Skeleton2Point in Table IV. Compared with other multi-branch methods using different modalities like RGB and depth, our Skeleton2Point achieves better performance and lower cost using skeleton and point clouds, demonstrating the validity and efficiency of modeling position relationships with point cloud methods in skeleton-based action recognition.

Ablation Study on Sampling and Grouping methods. To validate that the sampling and grouping methods (FPS+kNN) are suitable for the skeleton action recognition tasks, we also evaluate different sampling methods like random sample and grouping methods like ball query on NTU-RGB+D 60 dataset under the X-Sub setting with the joint input modality in Table V.

D. Comparison with Related Methods

Our proposed Skeleton2Point method is compared with the state-of-the-art methods on two different datasets: NTU-RGB+D 60 and NTU-RGB+D 120 datasets with the joint input modality to verify the competitive performance. Especially, only the joint stream dataset without bone and motion

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 06:21:34 UTC from IEEE Xplore. Restrictions apply.

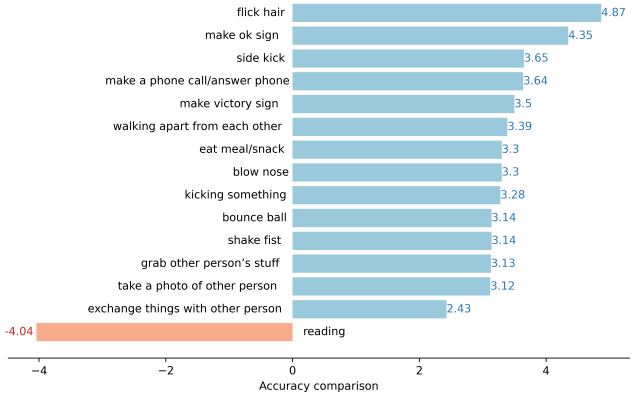


Fig. 3: Visualization of the top-15 actions with the highest change when the human skeleton branch is integrated with the point cloud branch for NTU-RGB+D 120 dataset under the X-Sub setting.

is used for a fair comparison because joints contain 3D coordinates the same as point clouds. If the bone or motion stream datasets were used, point cloud methods couldn't be well-compatible. Both the best results of all the methods shown in Table III are reported in the original papers. The comparison methods include GCN-based, Hyper-graph based, and transformer-based methods. Popular GCN-based methods (the upper 12 solutions in Table III) improve ST-GCN by constructing skeleton graphs with dynamic topologies to increase the receptive field of graph convolution. Recently proposed transformer-based methods (the last 5 solutions in Table III) treat each joint of the human skeleton as a token and use spatial and temporal self-attention operators to capture feature dependencies. Using a much more compact branch to model three-dimensional position relation and transformer-based network, our Skeleton2Point achieves comparable performance with the existing methods. Especially, in the most challenging NTU-RGB+D 120 under the X-Sub and X-Set settings, Skeleton2Point has reached state-of-the-art accuracy, which is extremely promising.

E. Visualization

Top-15 change analysis. To better illustrate the fusion effect of the human skeleton branch and point cloud branch, we visualize the top-15 actions with the highest change when the human skeleton branch is integrated with the point cloud branch on the most challenging benchmark, NTU-RGB+D 120 dataset under the X-Sub setting in Fig 3. SkeletonPoint (Ours) shows a substantial improvement in accuracy over the Skeleton branch baseline for all actions except the "reading" action. This observation proves that the introduction of point cloud information as an aid is beneficial for the model to further learn human actions. Secondly, point cloud information performs better on actions with larger motion amplitudes, where the accuracy of actions with more significant motions such as "flick hair" and "make ok sign" improves significantly, while the accuracy of actions with smaller motions such as "reading" decreases. An important reason for this change is that the point cloud information is sampled with FPS, making it easier for the point cloud model to capture the details of the motion and the overall trajectory. For motions with larger motion amplitude that have more dispersed feature points in the point cloud data, FPS covers a wider range of motion trajectories, thus improving the accuracy for motions with larger motion amplitude; In contrast, for actions with smaller motion amplitude, the point cloud density is relatively low, resulting in relatively fewer feature points sampled, which may then be overlooked.

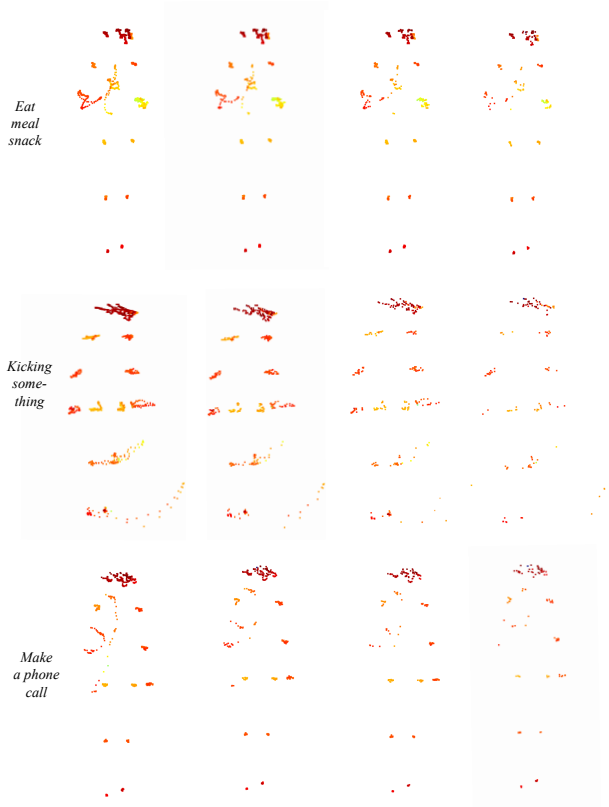


Fig. 4: Visualization of skeleton-based actions and corresponding point clouds.

Visualization. We visualize the down sample process during the point cloud learner extracting the spatial information of point clouds in Fig. 4. As for the visualization methods, we use matplotlib-3d for the skeleton and open3d for point clouds.

V. Conclusion

This paper proposes Skeleton2Point, a new representation learning framework for improved skeleton-based action recognition, which explores the learning mechanism of joints' position information by regarding joints as point clouds. Different from previous methods that neglect to model the positional information, our Skeleton2Point is the first to leverage point cloud methods into skeleton-based action recognition in a dual-branch approach. The method simultaneously demonstrates the validity of modeling position relationships with 3D coordinates in skeleton-based action recognition. Finally, the efficacy of our Skeleton2Point is well validated by thorough experiments on the NTU-RGB+D 60 and 120 datasets.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 06:21:34 UTC from IEEE Xplore. Restrictions apply.

REFERENCES

[1] M. Li, Z. Miao, X.-P. Zhang, W. Xu, C. Ma, and N. Xie, "Rhythm-aware sequence-to-sequence learning for labanotation generation with gesture-sensitive graph convolutional encoding," *IEEE Transactions on Multimedia*, vol. 24, pp. 1488–1502, 2021.

[2] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 14–29, 2015.

[3] C. Li, Z. Huang, Y. Yang, J. Cao, X. Sun, and H. T. Shen, "Hierarchical latent concept discovery for video event detection," *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2149–2162, 2017.

[4] Y. S., X. Y., and L. D., "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[5] K. L., P. K., and L. S., "Towards to-at spatio-temporal focus for skeleton-based action recognition," in *AAAI Conference on Artificial Intelligence*, vol. 36, no. 1, 2022, pp. 1131–1139.

[6] Z. Y., C. Z., L. C., F. Y., G. Y., X. X., and K. M., "Hypergraph transformer for skeleton-based action recognition," *arXiv:2211.09590*, 2022.

[7] C. He, C. Xiao, S. Liu, X. Qin, Y. Zhao, and X. Zhang, "Single-skeleton and dual-skeleton hypergraph convolution neural networks for skeleton-based action recognition," in *International Conference on Neural Information Processing*, 2021, pp. 15–27.

[8] Y. Zhu, G. Huang, X. Xu, Y. Ji, and F. Shen, "Selective hypergraph convolutional networks for skeleton-based action recognition," in *ACM International Conference on Multimedia Retrieval*, 2022, pp. 518–526.

[9] C. Y., Z. Z., Y. C., L. B., D. Y., and H. W., "Channel-wise topology refinement graph convolution for skeleton-based action recognition," in *International Conference on Computer Vision*, 2021, pp. 13 359–13 368.

[10] H. Liu, Y. Liu, Y. Chen, C. Yuan, B. Li, and W. Hu, "Transkeleton: Hierarchical spatial–temporal transformer for skeleton-based action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 8, pp. 4137–4148, 2023.

[11] X. W., M. Q., L. Y., L. R., P. C., and S. C., "Skeleton mixformer: Multivariate topology representation for skeleton-based action recognition," in *ACM International Conference on Multimedia*, 2023, pp. 2211–2220.

[12] Q. C.R., S. H., M. K., and G. L.J., "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *IEEE/CVF Computer Vision and Pattern Recognition*, 2017.

[13] Q. C.R., R. Z., Y. L., S. H., and G. L.J., "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Annual Conference on Neural Information Processing Systems*, vol. 30, 2017.

[14] G. Qian, Y. Li, H. Peng, J. Mai, H. Hammoud, M. Elhoseiny, and B. Ghanem, "Pointnext: Revisiting pointnet++ with improved training and scaling strategies," *Annual Conference on Neural Information Processing Systems*, vol. 35, pp. 23 192–23 204, 2022.

[15] H. Zhao, L. Jiang, J. Jia, P. H. Torr, and V. Koltun, "Point transformer," in *International Conference on Computer Vision*, 2021, pp. 16 259–16 268.

[16] L. Y., T. B., L. Y., L. L., and W. F., "Point cloud classification using content-based transformer via clustering in feature space," *IEEE/CAA Journal of Automatica Sinica*, 2023.

[17] X. Ma, C. Qin, H. You, H. Ran, and Y. Fu, "Rethinking network design and local geometry in point cloud: A simple residual mlp framework," *arXiv:2202.07123*, 2022.

[18] M. X., Z. Y., W. H., Q. C., S. B., L. C., and F. Y., "Image as set of points," *arXiv:2303.01494*, 2023.

[19] H.-g. Chi, M. H. Ha, S. Chi, S. W. Lee, Q. Huang, and K. Ramani, "Infogcn: Representation learning for human skeleton-based action recognition," in *IEEE/CVF Computer Vision and Pattern Recognition*, 2022, pp. 20 186–20 196.

[20] W. Xin, Y. Liu, R. Liu, Q. Miao, C. Shi, and C.-M. Pun, "Auto-learning-gcn: An ingenious framework for skeleton-based action recognition," in *Chinese Conference on Pattern Recognition and Computer Vision*, 2023, pp. 29–42.

[21] C. K., Z. Y., H. X., C. W., C. J., and L. H., "Skeleton-based action recognition with shift graph convolutional network," in *IEEE/CVF Computer Vision and Pattern Recognition*, 2020, pp. 183–192.

[22] L. Z., Z. H., C. Z., W. Z., and O. W., "Disentangling and unifying graph convolutions for skeleton-based action recognition," in *IEEE/CVF Computer Vision and Pattern Recognition*, 2020, pp. 143–152.

[23] C. Z., L. S., Y. B., L. Q., and L. H., "Multi-scale spatial temporal graph convolutional network for skeleton-based action recognition," in *AAAI Conference on Artificial Intelligence*, 2021.

[24] Y.-F. Song, Z. Zhang, C. Shan, and L. Wang, "Constructing stronger and faster baselines for skeleton-based action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 2, pp. 1474–1488, 2023.

[25] H. Zhou, Q. Liu, and Y. Wang, "Learning discriminative representations for skeleton based action recognition," in *IEEE/CVF Computer Vision and Pattern Recognition*, 2023, pp. 10 608–10 617.

[26] X. Wang, X. Xu, and Y. Mu, "Neural koopman pooling: Control-inspired temporal dynamics encoding for skeleton-based action recognition," in *IEEE/CVF Computer Vision and Pattern Recognition*, 2023, pp. 10 597–10 607.

[27] L. J., L. M., L. D., and L. S., "Hierarchically decomposed graph convolutional networks for skeleton-based action recognition," in *International Conference on Computer Vision*, 2023, pp. 10 444–10 453.

[28] J. Xie, Y. Meng, Y. Zhao, A. Nguyen, X. Yang, and Y. Zheng, "Dynamic semantic-based spatial graph convolution network for skeleton-based human action recognition," *AAAI Conference on Artificial Intelligence*, vol. 38, no. 6, pp. 6225–6233, Mar. 2024.

[29] Y. Zhou, X. Yan, Z.-Q. Cheng, Y. Yan, Q. Dai, and X.-S. Hua, "Blockgcn: Redefine topology awareness for skeleton-based action recognition," in *IEEE/CVF Computer Vision and Pattern Recognition*, 2024, pp. 2049–2058.

[30] J. Liu, C. Chen, and M. Liu, "Multi-modality co-learning for efficient skeleton-based action recognition," in *ACM International Conference on Multimedia*, 2024, pp. 4909–4918.

[31] X. Hao, J. Li, Y. Guo, T. Jiang, and M. Yu, "Hypergraph neural network for skeleton-based action recognition," *IEEE Transactions on Image Processing*, vol. 30, pp. 2263–2275, 2021.

[32] J. Wei, Y. Wang, M. Guo, P. Lv, X. Yang, and M. Xu, "Dynamic hypergraph convolutional networks for skeleton-based action recognition," *arXiv:2112.10570*, 2021.

[33] Z. C., H. Y., L. M., Yangand C., and H. X., "Skeletal spatial-temporal semantics guided homogeneous-heterogeneous multimodal network for action recognition," in *ACM International Conference on Multimedia*, 2023, pp. 3657–3666.

[34] Y. Zheng, H. Huang, X. Wang, X. Yan, and L. Xu, "Spatio-temporal fusion for human action recognition via joint trajectory graph," *AAAI Conference on Artificial Intelligence*, vol. 38, no. 7, pp. 7579–7587, 2024.

[35] W. Wu, C. Zheng, Z. Yang, C. Chen, S. Das, and A. Lu, "Frequency guidance matters: Skeletal action recognition by frequency-aware mixed transformer," in *ACM International Conference on Multimedia*, 2024, pp. 4660–4669.

[36] S. A., L. J., N. T., and W. G., "Ntu rgb+ d: A large scale dataset for 3d human activity analysis," in *IEEECVF Computer Vision and Pattern Recognition Conference*, 2016, pp. 1010–1019.

[37] S. J., Liuand A., P. M., D. G., Wangand L., and K. A.C., "Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 10, pp. 2684–2701, 2019.

[38] W. H., M. X., and L. Y., "Spatiotemporal multimodal learning with 3d cnns for video action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1250–1261, 2021.

[39] S. Das, S. Sharma, R. Dai, F. Bremond, and M. Thonnat, "Vpn: Learning video-pose embedding for activities of daily living," in *European Conference on Computer Vision*, 2020, pp. 72–90.

[40] B. XB., L. Y., Z. X., Z. S., and K. Chan, "Mmnet: A model-based multimodal network for human action recognition in rgb-d videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 3522–3538, 2022.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 06:21:34 UTC from IEEE Xplore. Restrictions apply.

[41] R. Ding, Y. Wen, J. Liu, N. Dai, F. Meng, and M. Liu, "Integrating human parsing and pose network for human action recognition," in *CAAI International Conference on Artificial Intelligence*, 2023, pp. 182–194.

[42] D. Ahn, S. Kim, H. Hong, and B. Chul Ko, "Star-transformer: A spatio-temporal cross attention transformer for human action recognition," in *Winter Conference on Applications of Computer Vision*, 2023, pp. 3319–3328.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 06:21:34 UTC from IEEE Xplore. Restrictions apply.